

APPLICATION
FOR
UNITED STATES LETTERS PATENT

APPLICANT NAME: William D. Corti et al.

TITLE: METHOD AND APPARATUS FOR ADDRESS DECODING OF
EMBEDDED DRAM DEVICES

DOCKET NO. FIS920010247US1

INTERNATIONAL BUSINESS MACHINES CORPORATION

ARMONK, NEW YORK

CERTIFICATE OF MAILING UNDER 37 CFR 1.10

I hereby certify that, on the date shown below, this correspondence
is being deposited with the United States Postal Service in an
envelope addressed to the Assistant Commissioner for Patents,
Washington, D.C. 20231 as "Express Mail Post Office to Addressee"
Mailing Label No. ET134022665US

on 8/28/01

Missy Shelly

Name of person mailing paper

Missy Shelly
Signature

8/28/01
Date

METHOD AND APPARATUS FOR ADDRESS DECODING OF EMBEDDED DRAM
DEVICES

BACKGROUND

The present invention relates generally to integrated circuit memory devices and, 5 more particularly, to a method and apparatus for address decoding of embedded DRAM (eDRAM) devices.

As a result of the package/pin limitations associated with discrete, self-contained devices such as DRAMs, memory circuit designers have used certain multiplexing techniques in order to access the large number of internal memory array addresses 10 through the narrow, pin-bound interfaces. Because these discrete DRAMs have been in use for some time, a standard interface has understandably emerged over the years for reading and writing to these arrays.

More recently, embedded DRAM (eDRAM) macros have been offered, 15 particularly in the area of Application Specific Integrated Circuit (ASIC) technologies. For example, markets in portable and multimedia applications such as cellular phones and personal digital assistants utilize the increased density of embedded memory for higher function and lower power consumption. Unlike their discrete counterparts, the eDRAM devices do not have the limited I/O pin interfaces with associated memory management circuitry. In fact, the typical I/O counts for eDRAM devices can number in the hundreds. 20

Another possible use for the newer eDRAM devices would be the ability to use them interchangeably with existing SDRAM modules, thereby forming a structure utilizing a standard SDRAM interface, but having an embedded memory module therein. Unfortunately, the memory array configurations for a standard SDRAM and an eDRAM are somewhat different. In addition, the existing multiplexing and addressing schemes 25 used by a memory manager (e.g., a hard disk controller) are not suited for use with an eDRAM structure, particularly with regard to the addressing of the array. For example,

5

the limited I/O pin interfaces in SDRAM memory management circuitry have resulted in the use of a “time shared” addressing method, wherein a row address is presented at an initial clock cycle and held in a register until the corresponding column address is presented at a later clock cycle. The eDRAM devices, on the other hand, do not make use of these time shared addressing techniques.

BRIEF SUMMARY

The above discussed and other drawbacks and deficiencies of the prior art are overcome or alleviated by a method for decoding a memory array address for an embedded DRAM (eDRAM) device, the eDRAM device being configured for operation with an SDRAM memory manager. In an exemplary embodiment of the invention, the method includes receiving a set of row address bits from the memory manager at a first time. A set of initial column address bits is then subsequently from the memory manager at a later time. The set of initial column address bits are translated to a set of translated column address bits, and the set of row address bits and the set of translated column address bits are simultaneously used to access a desired memory location in the eDRAM device. The desired memory location in the eDRAM device has a row address corresponding to the value of the set of row address bits and a column address corresponding to the value of the set of translated column address bits.

In a preferred embodiment, a first subset of the initial address bits is used to generate said translated column address bits, and a second subset of initial address bits is used to identify a specific location within an eDRAM column corresponding to the translated column address bits. The SDRAM memory manager processes memory address information in accordance with a first memory page structure, while the eDRAM device is configured in accordance with a second memory page structure. A memory page structure is defined by the number of columns included in a given row, and the number of storage locations located at each column in the given row. The first memory page

structure and the second memory page structure contain an unequal number of columns and an equal number of storage locations therein.

BRIEF DESCRIPTION OF THE DRAWINGS

Referring to the exemplary drawings wherein like elements are numbered alike in the several Figures:

Figure 1(a) is a three-dimensional representation of an existing 4Mb SDRAM device targeted for replacement with an equivalent embedded memory device;

Figure 1(b) illustrates an equivalent 4Mb embedded DRAM (eDRAM) device configured from two identical, 2Mb eDRAM modules;

Figure 2(a) is a three dimensional representation of a page structure for a given row the SDRAM device in Figure 1(a);

Figure 2(b) is a three dimensional representation of a page structure for a given row the eDRAM device in Figure 1(b), illustrating a reallocation of SDRAM column addresses, in accordance with an embodiment of the invention; and

Figure 3 is a block diagram illustrating an address decoding scheme for interfacing an SDRAM controller with the eDRAM array depicted in Figure 1(b), in accordance with an embodiment of the invention.

DETAILED DESCRIPTION

Referring initially to Figure 1(a), there is shown a three-dimensional representation of an existing 4Mb SDRAM device 100 targeted for substitution with an equivalent embedded memory device. As can be seen, the SDRAM 100 has a row depth of 1024 locations, along with a page structure of 256 columns x 16 bits. In contrast, Figure 1(b) illustrates an equivalent 4Mb embedded DRAM (eDRAM) device 102 which is configured from two identical, 2Mb eDRAM modules. A first eDRAM module 102a is

linked with a second eDRAM module 102b such that the second eDRAM module 102b is “stacked” behind first eDRAM module 102a.

It is readily apparent that the array structure of SDRAM 100 is different from the array structure of eDRAM 102. Although eDRAM 102 does have an identical row depth of 1024 locations, it also has a different page structure of 8 columns x 512 bits (256 bits each for first eDRAM module 102a and second eDRAM module 102b). Thus, a request for information stored in a particular row and column address in SDRAM format must be converted to a corresponding address location in the configured eDRAM device 102 shown in Figure 1(b).

Therefore, in accordance with an embodiment of the invention, a method and apparatus is disclosed wherein an embedded DRAM device may be configured for compatible use with existing SDRAM memory manager formats. Broadly stated, an initial row address presented by an SDRAM memory manager is held until a corresponding column address is presented at a later time. This scheme of delayed addressing is common with standard SDRAM hardware devices. Then, when the column address is received during a later clock cycle, the row and column addresses are combined and simultaneously applied to access the appropriate memory element within the eDRAM array. The column address bits initially received from the SDRAM memory manager are processed so as to result in a translated address location for the interfacing eDRAM device.

In order to more particularly illustrate the presently disclosed address translation scheme, reference may be made to Figures 2(a) and 2(b). In Figure 2(a), there is shown a three dimensional representation of a page structure 104 for a given row of the SDRAM device 100. Again, for a 4Mb SDRAM storage device, each of the 1024 rows will have 256 columns wherein 16 data bits may be stored at each address location. The columns are consecutively labeled 0-255. However, since there are only 8 columns in each row of eDRAM 102, the additional 248 columns that are recognized by the SDRAM control

hardware will have to be “reallocated” within the page structure of the eDRAM 102. This reallocation is illustrated in the eDRAM page structure 106 shown in Figure 2(b).

Since page structure 106 of configured eDRAM 102 allows the storage of 512 bits at each address location, this 512 bit storage capability is broken down into a series of 16 bit word slices or segments in each column. Thus, the amount of data stored in one column and row address in the eDRAM 102 will be equivalent to the amount of data stored in 32 columns of the SDRAM 100. As can be seen, the first 32 columns of the SDRAM page (0-31) are arranged from front to back in the first column (0) of the eDRAM page. The next 32 columns of the SDRAM page (32-63) are likewise arranged in the next column (1) of the eDRAM page, and so on.

It will be appreciated that this particular rearrangement of column locations from one memory structure to another represents only one possible embodiment thereof. Instead of arranging the SDRAM column address locations from front to back within each eDRAM column, the rearrangement could also have gone from side to side. As will now be described, the particular arrangement of reallocated SDRAM addresses in eDRAM affects the handling of the address bits received from an SDRAM controller.

Referring now to Figure 3, there is shown a block diagram illustrating an address decoding scheme for interfacing an SDRAM controller (e.g., an external memory manager) with the eDRAM array depicted in Figure 1(b). Again, due to the differences in array configurations as described above, a request for information in a particular row and column format will be converted to a corresponding location in the eDRAM array. Initially, the row address bits from a memory manager 200 are received and stored in a register 202. In standard SDRAM controllers, I/O pin limitations typically result in the complete transfer of address information bits over more than one clock cycle. On the other hand, with the speed of eDRAM, each access thereto can be accomplished in one clock cycle, even if it falls in a different eDRAM column.

Because there are 1024 (2^{10}) rows in the both the SDRAM and eDRAM array configurations, the 10 address bits (designated RA9-RA0) will be processed as is. In contrast, the 256 (2^8) column format recognized by SDRAM hardware means that there will be 8 column address bits (designated CA7-CA0) presented by memory manager 200 at a subsequent clock cycle. These bits are then loaded into a counter 204. The three most significant bits (7, 6 and 5) will join the ten row address bits as the translated eDRAM address in a broadside address register 206. Bit 4 will represent a selection bit for determining which of the two (2Mb) eDRAM modules 102a, 102b to access, and is fed from counter 204 to steering logic 208. Finally, the remaining four lowest order bits (3, 2, 1 and 0) are fed to a multiplexing device 210 for accessing the correct word slice within a 512 bit eDRAM column.

The specific distribution of the original column address bits stored in counter 204 will be understood with reference once again to Figure 2(b). The value of the four least significant bits (3, 2, 1 and 0) in the originally presented column address sent to multiplexing device 210 determine which 16-bit word slice in a given column of a given eDRAM module will be accessed. For example, column number 0 in the SDRAM format corresponds to the original column, eight-bit binary address (00000000). Counting forward, it can be seen that the first 16 addresses (00000000 - 00001111) correspond to the first 16 word slices in column number 0 of the eDRAM array 102. Then, for the next sequential address (00010000), the lowest four SDRAM column address bits have reset to 0000, but the fifth bit (bit #4) is now 1. This means that SDRAM column address number 16 is located at the first word slice of the first column of the second eDRAM module 102b. Therefore, the fifth SDRAM column address bit (bit #4) determines which of the two eDRAM modules are accessed.

The next 16 word slices (addresses 00010000 - 00011111) complete the assignment of addresses in eDRAM column 0. Address 00100000 (SDRAM column number 32) is then assigned to the front of column number 1 in the eDRAM array. As

this same pattern is completed through all 256 SDRAM column addresses, it becomes evident that the three most significant SDRAM column address bits (7, 6 and 5) determine which of the eight column locations in the eDRAM array is accessed.

For an illustrative example of the above described addressing scheme, it will be assumed that the SDRAM memory manager 200 is requesting access of stored information in row number 0, column number 95 of a 4Mb (1024 row x 256 column) SDRAM memory array. The ten row address bits initially received into register 202 are, accordingly, (0000000000). These ten row address bits are then held in register 202 until the column address bits (in SDRAM format) are received from memory manager 200 into counter 204 during a later clock cycle. The binary value of the column address bits corresponding to column number 95, therefore, is (01011111).

Referring again to Figure 2(b), it is seen how the present addressing scheme translates the initially transmitted column address bits in SDRAM format to a correctly reallocated column location in the eDRAM array. The three most significant bits (010) represent the translated column address for the eDRAM array and join the ten row address bits in broadside address register 206. Thus, column number 95 of the SDRAM array will correspond to column number 2 (binary 010) of the eDRAM array. The next bit in the SDRAM column address, being the eDRAM module selection bit, has a value of "1", signifying that column number 95 will further be found in column number 2 of the second (rearward) eDRAM module. Finally, the four least significant bits (1111) pinpoint the correct word slice in column 2 of the eDRAM array. As can be seen in Figure 2(b), column number 95 is located at the last (sixteenth) word slice in the second column of the second eDRAM module 102.

Through the use of the above disclosed address translation scheme, it is seen how embedded DRAM devices may be made compatible with existing SDRAM products, without necessarily having to redesign the industry standard hardware associated with the SDRAM products. Although a 4Mb memory device is depicted, the particular memory

5

size is exemplary only as the present invention embodiments are also applicable to various memory sizes. Furthermore, the principles of the invention may also be applied to a memory device that has a different row configuration, as well as a column configuration. In such a case, additional translational logic would be implemented to convert a row address in one memory configuration to the corresponding row address in another memory configuration.

10

While the invention has been described with reference to a preferred embodiment, it will be understood by those skilled in the art that various changes may be made and equivalents may be substituted for elements thereof without departing from the scope of the invention. In addition, many modifications may be made to adapt a particular situation or material to the teachings of the invention without departing from the essential scope thereof. Therefore, it is intended that the invention not be limited to the particular embodiment disclosed as the best mode contemplated for carrying out this invention, but that the invention will include all embodiments falling within the scope of the appended claims.

15